DOCUMENT RESUME

ED 240 498 CS 007 427

AUTHOR Russ-Eft, Darlene F.; McLaughlin, Donald H.
TITLE Ideas for Reading and Writing Courseware: Needs and Development Opportunities for Educational Computer Software in Reading, Writing, and Communication Skills. Executive Summary.
INSTITUTION American Institutes for Research in the Behavioral Sciences, Palo Alto, Calif.
SPONS AGENCY Office of Educational Research and Improvement (ED), Washington, DC.
REPORT NO AIR-26501-ES-83
PUB DATE Jun 83
CONTRACT 400-82-0021
NOTE 13p.; For the full report, see CS 007 428.
PUB TYPE Viewpoints (120) -- Information Analyses (070) -- Reports - Evaluative/Feasibility (142)

EDRS PRICE MF01/PC01 Plus Postage.
DESCRIPTORS Academic Standards; Communication Skills; *Computer Assisted Instruction; *Computer Programs; Cooperation; Educational Innovation; Educational Objectives; Evaluation Criteria; Incentives; *Program Development; *Reading Instruction; Reading Skills; *School Business Relationship; Speech Communication; Teacher Role; *Writing Instruction; Writing Skills

ABSTRACT

More effective reading, writing, and communication courseware can be promoted through clearer guidelines in courseware development, stronger incentives for courseware developers, and greater motivation for teachers to seek out and use the courseware. In reading and writing, software guidelines must reflect traditional instructional objectives--or functional domains--needed information processing skills, and courseware implications. Incentives for developers include assurance of a large and stable educational market and more effective measures to control software piracy. Teacher resistance, a major obstacle to the successful implementation of computer assisted instruction, can be countered through improved teacher training, inclusion of positive teacher roles in the lessons, user friendly materials, and the establishment of courseware evaluation standards. Recommendations for software developers, schools, and state and federal policy makers include creating computer literacy standards in preservice teacher education; establishing local teams of teachers, programmers, publishers, and researchers to generate high quality courseware; and promoting strong school positions against software piracy. (MM)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED240498

AIR/26501/ES/83

AMERICAN INSTITUTES FOR RESEARCH
IN THE BEHAVIORAL SCIENCES
P.O. Box 1113, 1791 Arastradero Rd., Palo Alto, Ca. 94302 ● 415/493-3550

U.S. DEPARTMENT OF EDUCATION
NATIONAL INSTITUTE OF EDUCATION
EDUCATIONAL RESOURCES INFORMATION
CENTER (ERIC)
☐ This document has been reproduced as received from the person or organization originating it.
☒ Minor changes have been made to improve reproduction quality.
• Points of view or opinions stated in this document do not necessarily represent official NIE position or policy.

EXECUTIVE SUMMARY

IDEAS FOR READING AND WRITING COURSEWARE

Needs and Development Opportunities for Educational Computer Software in Reading, Writing, and Communication Skills*

Darlene F. Russ-Eft
Donald H. McLaughlin

American Institutes for Research
P. O. Box 1113
Palo Alto, California 94302

June 1983

* This work was supported by a contract from the Office of Educational Research and Improvement of the U.S. Department of Education, contract #400820021. Any opinions, findings, conclusions, or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the Department of Education.

Reports of this project include the following:

Russ-Eft, D. F., McLaughlin, D. H., & Elman, A. Issues for the development of reading and writing software. Palo Alto, CA: American Institutes for Research, 1983.

Russ-Eft, D. F., & McLaughlin, D. H. Ideas for reading and writing courseware. Palo Alto, CA: American Institutes for Research, 1983.

CS007427

LIST OF APPENDICES

APPENDIX A.	Idea Work Group Participants	127
APPENDIX B.	Listing of Courseware by Language Skills Area	135
APPENDIX C.	School Microware Evaluation Form	151
APPENDIX D.	Evaluation Questions: School Courseware Journal	153

INTRODUCTION

Recent surveys of elementary and secondary schools reveal that the number of microcomputers in the schools has doubled in the past year and that the growth in equipment purchases continues. During the same period, advances in the quality of computer hardware have been extremely rapid. In contrast, advances in educational software, particularly in the areas of reading, writing, and communication skills, have been slow.

Reading, writing, communication, and computation are the fundamental basic skills which schools are expected to teach to American youth. But a cursory review of the many catalogs, advertisements, and lists of courseware reveals that the availability of software in all areas except science and math lags far behind any reasonable expectation of those in schools that have acquired hardware. Furthermore, the vast majority of existing reading, writing, and communication courseware focuses on the simplest of tasks--spelling and vocabulary. This is true in spite of the national spotlight on declining reading scores and especially the decline in higher levels of reading and writing skills.

There are three major causes of this state of affairs. First, innovations designed to meet traditional needs of teachers in this area, such as grading of compositions, are very difficult, if not impossible, to implement with existing hardware. Second, the computer medium is not as directly related to the subject matter in reading, writing, and communication as in math and science. As a result, innovative ideas are needed in order to apply the computer's capabilities to meet the needs of teachers and students for improvements in reading and writing instruction. Finally, the audience of reading, writing, and communications teachers is not yet prepared to compete with math and science teachers for computer resources.

A critical need exists for understanding the nature of the barriers that limit the availability of high quality software for reading, writing, and communication skills. Equally important is the need for an examination of possible approaches that teachers, developers, publishers, and educational administrators can employ to remove the barriers.

PROCEDURES

The Office of Educational Research and Improvement, U.S. Department of Education, awarded a contract to the American Institutes for Research (AIR), in Palo Alto CA, to examine the needs and opportunities for reading, writing, and communication software. Rather than carry out a broad evaluation of available courseware, which would duplicate other efforts, the AIR staff identified critical problem areas and gathered information necessary to explore potential solutions to them.

As background, the project prepared an issue-oriented literature review, including research on reading and writing, evaluations of the effects of computer-assisted instruction, software evaluations, and clearinghouse materials on computers in instruction. The review addressed needs for courseware guidelines in reading and writing, for support for developing high quality software, and for teachers' acceptance of computers in reading and writing instruction. Its results were discussed by a National Advisory Panel of experts in reading, writing, software development, and software publishing.

To clarify and explore solutions to the problems identified in the literature review, a series of 17 Idea Work Group meetings were held throughout the United States--in Bedford TX, Cambridge MA, Dallas TX, Gainesville FL, Glenview IL, Melrose Park IL, Osseo MN, Palo Alto CA, Richmond VA, Tallahassee FL, and Washington DC. The participants included 34 teachers, 25 school administrators, 14 teacher trainers, four state and three federal education officials, 22 researchers, 22 software developers, 13 publishers, and eight hardware manufacturers. The focus of each of these all-day meetings was to examine the barriers to software development in reading and writing skills and, through use of creative problem-solving techniques, to explore solutions for these problems.

In addition to conducting the Idea Work Groups, AIR contacted over 200 developers and reviewed 12 major software catalogs and the 1981, 1982, and 1983 issues of eight computer magazines and two computer news-weeklies to identify relevant software. The software packages were categorized by reading or writing skill area (using a recent basal reading series). This categorized listing helped to identify skill areas needing more attention.

Finally, to supplement the coverage of software developers in the Idea Work Groups, the staff held private discussions with 40 other software developers, to assess general perceptions about the educational software market and to identify additional problem areas.

RESULTS

Progress in the areas of reading, writing, and communication courseware will be facilitated by (1) clearer guidelines for the design of good courseware, (2) increasing the incentives for the development of reading, writing, and communications courseware, and (3) increasing the incentives for teachers to seek out and use the courseware.

(I) Clearer guidelines--How must traditional reading, writing, and communication objectives be translated in order to convert them into courseware that makes use of the computer's potential?

(II) Developers' incentives--How can developers and publishers be assured that they can recover their investment--through knowledge of the school market and through the prevention of software piracy?

(III) Teachers' incentives--How can teacher training be implemented to overcome resistance, how can courseware be designed to incorporate new teachers' roles and to be more friendly to teachers of reading, writing, and communication skills, and how can evaluation standards incorporating teachers' needs be developed and used?

I. CLEARER GUIDELINES

Overall, when reading, writing, and communication courseware have been compared with mathematics courseware, the former have proven less effective. This is not surprising given the greater difficulty of developing usable courseware focusing on reading, writing, and communication skills. Clearer guidelines for all communications courseware are needed. Objectives that have been taught as comprising the ability to read or write, especially at the higher levels, are defined without reference to the underlying skills that must be acquired to exhibit the ability.

Reading

Reading involves extracting information from text, where that information is not only printed words and sentences but also combinations involving text and pictures, diagrams, graphs, and so forth. Reading integrates of a variety of skills; some are controlled by the surface text (or "bottom-up" skills), and some involve conceptual processes (or "top-down" skills).

The basic message of recent reading research is to broaden the concepts of reading instruction and to identify skills that differentiate good from mediocre readers, going beyond the simple decoding tasks. In order to translate this message into software guidelines, we must identify particular links between reading tasks and learning sequences that improve skills needed for those tasks. The link involves analysis of three levels: (1) functional domains, (2) information processing skills, and (3) courseware implications.

1. Functional domains. There are "traditional" instructional objectives that many teachers feel are important, such as: reading readiness, phonetic analysis, vocabulary, recognizing memory words, speed reading, study skills, comprehension, and structural analysis. Although an information-processing approach based on new research may facilitate the design of high-quality software, the developer must also convince teachers of the usefulness of the resulting product in their instructional programs. In order to communicate with teachers and convince them of the usefulness of a piece of software, any new skill domain should be compared with those tasks (and instructional objectives) that characterize traditional taxonomies.

2. Information processing skills. As a part of any analysis of information processing requirements, it would be useful to focus on three stages of the overall reading process: before the text is read; while the text is being read; and after the eyes have left the page. Each of these stages requires a different set of information processing skills. Instructional decisions before reading focus on vocabulary and conceptual background needed for a particular text. Instructional decisions during reading emphasize the anticipation of structural, organizational, and rhetorical elements and sensitivity to cohesive aspects of the text. Instructional decisions after reading focus on post-questions, feedback, and textual recall.

3. Courseware implications. Types of exercises that would be useful to include in courseware on a specific reading skill clearly depend on one's conception of that skill. Some suggestions for courseware from the literature and from the Idea Work Group participants are:

- Vary audience, author, or voice to help students become aware of these shifts.
- Provide concept and language awareness activities prior to reading to help students think about what they already know about a specific topic. Have them anticipate what they will read in the text.
- Use color to represent sounds within a phonically-based system. As children learn the sounds, the computer could gradually remove the color distinctions.
- Present interactive texts in which the reader can request supplementary information, such as a vocabulary item, a graphic display relevant to the text, or the main idea of the paragraph.
- Create an adventure game in which the child receives written instructions on how to get past a series of obstacles. These could get more and more complicated with a sequencing of skills.

Writing

"Writing", the middle "R", covers a broad range of tasks and an even broader range of component skills and processes. On the one hand, it covers the "mechanics" of conforming to the conventions for producing visual language; on the other, it covers the creative processes involved in communicating ideas. Although software can easily be generated to provide instruction in the former, it is in the latter area that the most exciting opportunities for computer-assisted learning lie.

1. Functional domains. Writing skills can be classified in three ways: developmental sequence, task purpose, and task phase. One traditional developmental scope and sequence chart for writing contains six major headings of objectives: written expression, (including sentence writing, paragraph writing, report writing, and proofreading), capitalization and punctuation, usage (including verbs, modifiers, pronouns, and common confusers), structure, spelling and encoding, and handwriting. As in the area of reading, these traditional category labels suggest to teachers ways in which a software package can be a useful addition to the classroom. However, these categories focus on sets of rules and patterns to commit to memory and are quite different from the information-processing and cognitive skill categories that researchers have recently been exploring.

2. Information processing skills. There are two types of skills underlying writing, discrete rules to be learned and skill sub-domains that are gradually mastered. In both cases, these span the developmental, purposive, and phase categories. There are nine types of rules to be learned, ranging from the forming of letters and spelling to rules for generating structure and content. There appear to be eight skill sub-domains, ranging from vocabulary to problem-solving management to memory.

3. Courseware implications. Two types of courseware can be distinguished: skill objectives-based and general writing support. In the project's Idea Work Groups, many ideas for software to assist in the teaching of writing were offered, ranging from global principles to specific programs. Twelve general suggestions for developers were extracted from the transcripts of the meetings. A few of these ideas are included in the following excerpts from the meetings.

- Use an easy word-processing system in the drafting and revising process. It is important to convey the idea that, when you write, you don't just use your first draft. Rather, there is always room for improvement.

- Include some "canned" prompts in the word processing system to guide students' writing of a story: What is your setting; what is your initiating event; what is the goal?

- Include optional spelling and grammar correctors. (This is not so easy because grammar overlaps into semantics and requires a processor that can recognize semantic as well as syntactic transformations.)

II. DEVELOPERS' INCENTIVES

Anybody with a few hours training in BASIC can quickly write a program that will present a list of correctly spelled and misspelled words and give students scores and feedback for correct identification of the misspellings. Developing a program to teach spelling skills effectively is an entirely different project, involving analysis of cognitive skills, drafting programs to teach the identified skills, tryouts with children, revision and debugging, and preparation of documentation and supplementary materials. However, few individuals or publishers are prepared to expend the effort needed for such an approach without good assurance of a return on their investment, which is dependent in part on the size and characteristics of the school market and on the prevention of software piracy.

School Market

Determination of the return on investment depends heavily upon the size of the market. But, for developers and publishers considering work in the educational market, other factors are also critical. Generally, the following create problems for developers and publishers:

- lack of knowledge about the future of microcomputers in the schools,
- existence of a wide-range of computers with a lack of compatible systems, and
- lack of market guides that identify and describe actual school software purchasers.

For these reasons, some developers and publishers are focusing on the business and the home markets while they wait to see what happens in the schools. As was stated by one Idea Work Group participant, "there has not been enough unanimity in education. Therefore, we have invested in business software. If the schools could come to some consensus about what they would like, then we might be willing to make some investments in software developments for public school systems. This is no doubt true for other developers and publishers."

Software Piracy

There are roughly 3,000,000 children at each grade level in the 100,000 elementary and secondary schools in the country. At one extreme, where each courseware unit is usable by one student alone, a 10% market penetration would yield roughly 300,000 sales per year; this would allow inexpensive pricing for courseware requiring even hundreds of thousands of dollars to develop. If a single copy of a courseware unit were bought by each school and copied or networked, however, the corresponding estimate (over all years!) for that unit would be reduced to fewer than 10,000; this would require expensive pricing to recover several hundred thousand dollars of development costs, when added onto per-unit production, distribution, and marketing costs.

What solutions are there to the problems created by copying and networking? For unauthorized copying (otherwise known as "piracy" and "theft"), there appear to be three broad categories of solutions.

1. Prevention of copying. Copyright laws exist but copying persists. Basically, as long as courseware is transmitted on floppy disks, the only barrier to extremely inexpensive copying is the use of nonstandard formatting (e. g., writing "between the tracks" on a diskette). However, sophisticated "lock-picking" programs are likely to be continually improved to defeat the use of nonstandard formatting. Alternatives to the use of nonstandard formatting of disks that might be considered for preventing copying include:

- o innovative approaches designed to convince teachers not to use illegal copies;
- o hardware solutions such as storage of courseware on PROMS or other less copyable media; and
- o inclusion of supplemental material with the courseware (such as a workbook or a keyboard overlay) that is either "used-up" or at least not so cheaply copied or provision of special services with the courseware purchase.

2. Subsidizing development. This approach involves paying for the courseware development "up front," by some agency or foundation, rather than through standard publishing agreements and royalties. Its primary disadvantage is that little incentive exists to improve the courseware beyond initially delimited specifications. Furthermore, the free marketplace cannot operate to select the most effective courseware on the basis of performance.

3. Institutional purchase agreements. If sales of courseware can be made in sufficiently large blocks to state education agencies or large local education agencies, provisions could be included in the terms that could serve to decrease the effects of copying, for example licensing of multiple copying of disks for a fee. Indeed, with an expected rise in the use of local-area networks to link large numbers of microcomputers together within a school or district, institutional lease/purchase or license agreements may become the preferred solution. The key to the success of this solution is that the pressure on public institutions to conform to copyright laws is substantially greater than the pressure on individuals.

III. TEACHERS' INCENTIVES

A major barrier to the implementation of computers in instruction, particularly in reading and writing, is that of teacher resistance. Studies of the implementation of computer-assisted instruction have indicated that one of the greatest obstacles to implementation is that of low teacher acceptance. Teacher resistance stems from several different concerns about the technology: fear of mechanical problems, concerns about the costs of the equipment and software,

fear of being replaced by the computer, lack of time, fear that expanding technology signifies progressive dehumanization, and the perception that the computer is not appropriate for teaching reading and writing skills.

Therefore, guidance must be made available both to educators interested in promoting computer use among their colleagues and to potential courseware developers concerned about structuring their products to foster teacher enthusiasm. There are four broad categories of strategies to consider in promoting teacher acceptance of courseware: (1) teacher training, (2) inclusion of positive teacher roles in the lessons, (3) care in ensuring user friendliness, and (4) development of courseware evaluation standards.

Teacher Training

Increased knowledge and information about microcomputers may dispel certain teacher fears. Various authors, in discussing strategies for overcoming resistance to microcomputers, identify the need for teachers to be better informed about computers and their potential use in the classroom. A common suggestion is to hold inservice programs with workshops and presentations to demonstrate the microcomputer's capabilities. It must be recognized that such in-service training must be conducted at a level appropriate to the needs and experiences of the teachers involved.

1. Inservice training for "beginners." Inservice training for starters should focus on the fears and misconceptions leading teachers to avoid computers. As one teacher stated, "there is a need for a 'therapy' session in which philosophical concerns are brought up." Teachers need to "realize that they can plug in the computer and play with it without worrying that they are going to break it." In addition, these sessions need to give teachers hands-on experiences with computers. They need to be shown how to run a program and be given the opportunity to run it themselves. Allowing teachers complete access to the machines will encourage them to have a sense of personal control.

2. Inservice training for "users." At this level, the teachers have overcome their initial fears of the computer and are interested in some level of implementation in the classroom. The training at this level must deal with different options for computer use, with software evaluation, and with classroom management.

3. Inservice training for teacher/developers. Criticisms have been raised concerning the focus of many computer training programs for teachers. Many times the instructor is a computer specialist who focuses the instruction on computer programming. But, most teachers do not want to learn programming; they want to learn how to use the hardware and the software. There are, however, a few teachers who quickly become quite facile with computers. They need a course to help them learn how to design and implement their ideas through software.

In the initial stages, this training can focus on the use of authoring systems and higher level languages. Some teachers will even want to go beyond the limitations of these languages. In both cases, teachers need to have access to experts to provide answers to their specific problems.

Teacher Roles

A critical failing of much of the existing courseware is the lack of consideration of problems raised by the introduction of the courseware into the overall instructional program for students. The essence of the concept of including a role for the teacher in a courseware unit is that the courseware should be implemented in different ways for different students in different settings.

There are two approaches to allowing for teacher variation in software: (1) including provisions for teachers to make choices in the design of the lesson, and (2) including provisions for teachers to provide crucial input during the lesson (including overnight inputs in multi-day lessons). Provisions for teachers to alter the design of lessons vary along a continuum from that of making small alterations all the way to actually designing the software. The second type of provision, allowing for teacher inputs during the lesson, presents a very challenging problem; on the surface, it appears to defeat one of the strengths of the computer--its ability to attend to one child completely while the teacher focuses on the remainder of the class. It seems clear that the form of the teacher's input should be designed so that it can be made when the teacher has time available. For example, a word processor might assist in writing instruction by indicating spelling or grammatical errors to a student directly. The problem of indicating the need for better transitions, however, could be left to the teacher, who would examine a composition while the student is engaged in other activities.

User Friendliness

User friendly software is critical for teacher acceptance. And, what does user friendly mean? "A program is user friendly if a fifth grade teacher can try it out without having to provide elaborate instructions." "User friendly software is self-directing. It provides a self-explanatory menu that directs the user to the available options. The documentation, if needed at all, is completely understandable; it avoids computer jargon. And, above all, the program is dependable."

Three basic approaches to user friendliness are: (1) avoiding requirements that the teacher or student perform actions that he or she finds difficult or annoying; (2) providing products that eliminate chores teachers traditionally find annoying; and (3) including components in the courseware that correspond to relevant preferences of teachers.

Courseware Evaluation Standards

There are a wide variety of evaluation methods, and reviews of the same courseware are often contradictory. In order to select high quality courseware, schools need criteria and procedures that are easy to use and ensure the selection of the best available programs to meet their needs. Courseware evaluations should be based on instructional objectives, evidence of usability and effectiveness, user friendliness (including freedom from "bugs" and from annoying limitations), and clarity, accuracy, and completeness of documentation, as well as price.

RECOMMENDATIONS

A large set of useful recommendations, for software developers, for schools, and for federal and state policy-makers, were contributed by the participants in the Idea Work Groups and in the National Advisory Panel meetings. Controversies were discussed, and the recommendations were refined accordingly. The following summarizes the most important recommendations of the project.

1. The federal government should develop a leadership position, in order to focus attention of software developers on the school market and in order to ensure excellence and equity in the use of computers in education.

2. The federal government should fund projects aimed at the development of high quality software, based on careful research. Development costs are high, and the school market is otherwise not sufficiently large to attract the needed developers' efforts.

3. The federal government should support efforts in the areas of developing (a) standardization in hardware and in software languages, (b) software evaluation standards, and (c) regional resource centers, in order to increase schools' abilities to select and acquire high quality software.

4. The federal and state governments should support hardware acquisition by schools, such as through tax credits to manufacturers. The size of the hardware base is critical for the development of high quality software.

5. States should establish standards for computer literacy in the preparation of new teachers, in order to alleviate the need for massive in-service training.

6. Schools should develop long-term plans for use of computers, including hardware and software acquisition procedures and in-service training at three different levels of expertise: first-time users, continuing users, and potential developers.

7. Local teams of teachers, programmers, publishers, and researchers should be formed in order to generate high quality, relevant courseware. The responsibility for forming these teams might lie with any of the constituents, but the value of forming the teams is unquestionable.

8. Courseware developers should make use of research on reading and writing in order to generate exercises and activities that require practice of appropriate skills; and they should describe courseware in terms familiar to teachers.

9. Schools should require that courseware be of high quality, with (a) documentation that includes suggested uses, (b) mechanisms for adaptation to local classroom variations, and (c) accurate use of English.

10. Schools should take strong positions opposed to the use of illegally copied software by students and staff. Software piracy is viewed by publishers as a significant deterrent to investment in the development of high quality courseware.